# Variance calculation

## Conceptual design

This calculates the variance inherent in the calculated mass, based on of the buffer of voltage values. This buffer forms a ‘population’ of finite size, pre-defined as VBUFSIZE = 8.

Thus the general formula for population variance is applied:

for number of samples , individual samples and mean . Here VBUFSIZE, is each voltage sample in the buffer, and is the mean voltage. This mean voltage is already calculated by a separate function so it can be easily accessed by the variance function.

The decision was made to take the variance of the voltage buffer and then convert it to a mass value, rather than taking the variance of several calculated mass values, because our setup removes the need to store a separate buffer of mass values.

Floating point arithmetic is used to provide an accurate value of the deviation, which is usually much smaller than the actual mass value. The voltage values are stored as unsigned integers, but are cast to floats for the mathematical operations to avoid truncation, then back to an unsigned integer for reporting at the end.

The variance in voltage is then converted into a variance in mass, for reporting to the user, by using the same calibrated mass-voltage conversion as for the weight. However, there is an important difference. Because the variance deals only with differences in voltage rather than absolute values, there is no need to apply an offset; only the gradient in the mass-voltage line is necessary. As an example, consider a perfectly stable voltage which thus has a variance voltage of 0. Passing this into the voltage-to-mass calculation will result in , giving a non-zero variance mass when the answer should be zero. Instead, for variance calculations the correct formula is only. This is implemented with a flag, as explained in the ‘Software design’ section.